W3C®

XP-002297512

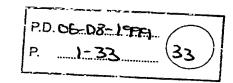P.D. 06-08-1999
P.   1-33   (33)

NaVigation Markup Language (NVML)

W3C Note 6 Aug 1999

This Version:
>   http://www.w3.org/TR/1999/NOTE-NVML-19990806

Latest version:
>   http://www.w3.org/TR/NVML

Authors:
>   Minoru Sekiguchi, Fujitsu Laboratories Ltd., minoru@flab.fujitsu.co.jp
>   Kuniharu Takayama, Fujitsu Laboratories Ltd., taka@iias.flab.fujitsu.co.jp
>   Hirohisa Naito, Fujitsu Laboratories Ltd., naitou@flab.fujitsu.co.jp
>   Yoshiharu Maeda, Fujitsu Laboratories Ltd., haru@iias.flab.fujitsu.co.jp
>   Hisayuki Horai, Fujitsu Laboratories Ltd., horai@iias.flab.fujitsu.co.jp
>   Masahito Toriumi, Fujitsu Limited, toriumi@sysrap.cs.fujitsu.co.jp

Status of this Document

This document is a submission to the World Wide Web Consortium from Fujitsu Laboratories Ltd. (see Submission Request, W3C Staff Comment).

This document is a NOTE made available by W3C for discussion only. This indicates no endorsement of its content, nor that W3C has had any editorial control in its preparation, nor that W3C has, is, or will be allocating any resources to the issues addressed by the NOTE.

Abstract

A variety of mobile information appliances are developed rapidly such as smart phones with capability of Internet access, PDAs (Personal Digital Assistants) equipped with GPS (Global Positioning System), and car navigation systems. A navigation service is one of the indispensable services to these mobile information appliances, but currently it is only available with car navigation systems. In order to use the navigation service not only on cars but also on trains, on buses, and on foot with various mobile information appliances, a common data format for describing navigation information such as locations of points and route information is required. The "NVML" (NaVigation Markup Language) is a markup language for describing the navigation information satisfying the above requirement. The NVML enables us to use a navigation service at any time and anywhere, and it will also be used in other services such as a guide service for sightseeing, a travel planning service, and a publishing service for tourists.

Contents

1. Introduction

1.1. What is NVML?

A variety of mobile information appliances are developed rapidly such as smart phones with capability of Internet access, PDAs (Personal Digital Assistants) equipped with GPS (Global Positioning System), and car navigation systems.

A navigation service is one of the indispensable services to these mobile information appliances, but currently it is only available with car navigation systems. In order to use the navigation service not only on cars but also on trains, on buses, and on foot with various mobile information appliances, a common data format for describing navigation information such as locations of points and route information is required.

21-09-2004

The "NVML" (NaVigation Markup Language) is a markup language for describing the navigation information satisfying the above requirement, and is defined by using the XML (Extensible Markup Language) authorized by the W3C (World Wide Web Consortium).

Since the contents of NVML data (ex., Kyoto-Nara Tour) are marked up by tags (ex., <title>), anyone can understand the data easily and computers can process and search the data quickly. Besides, NVML has a capability to describe several types of navigation information such as:

- a route from the current point to a destination point;
- a way to a shop from the nearest station;
- transportation courses;
- sightseeing courses;
- tour schedules.

The NVML enables us to use a navigation service at any time and anywhere, and it will also be used in other services such as a guide service for sightseeing, a travel planning service, and a publishing service for tourists.

This document describes the NVML specification. In the following, the system that inputs and/or outputs any NVML data is called an NVML system.

1.2. Features

The NVML has the following features:

- XML-Base
  - o NVML data can be wide-spreaded through webs as an XML application.
  - o NVML users can use various XML tools.
  - o The NVML inherits the XML features such as readability and processability.
- Route Navigation and Point Guidance
  - o The NVML has two functionalities:
    route navigation, i.e., an information service along route, and
    point guidance, i.e., an information service around point of interest.
- Several Output Information Types
  - o The NVML can deal with several output information types including:
    maps, text characters, voices, and images.
- Authorized by Anyone
  - o NVML data can be authorized by anyone.
  - o For example, users write their driving courses, and tour companies make their tour data.
- Released by Anyone
  - o NVML data can be released by anyone.
  - o For example, users can mail an NVML data to their friends, and
    contents providers can serve various NVML data from their sightseeing database.
- At any Time and Anywhere
  - o NVML data can be used at any time and anywhere.
  - o For example, users will be able to use a navigation service at home, at office, during moving on foot, on bus, on train with various information appliances, for example, car navigation systems, personal computers, PDAs (Personal Digital Assistants), and cellular phones.
- Actual Navigation/Guidance and Simulation
  - o NVML data can be used in actual navigation/guidance as well as in virtual

simulation.

2. Specification

2.1. File Extension Suffix

The file extension suffix of NVML files is ".nvml" and ".nvm". These suffixes are used to recognize whether the file is an NVML file or not.

2.2. Media Type

MIME (Multipurpose Internet Mail Extensions) Media Type [1] of NVML files is "application/x-nvml".

2.3. XML Declaration

The XML declaration of NVML data is as follows:

```
<?xml version="1.0" encoding="xxx"?>
<!DOCTYPE nvml SYSTEM "http://yyy/nvml.dtd">
```

The value of the attribute "encoding" (i.e., "xxx" in the above description) is subject to the XML (Extensible Markup Language) specification [2]. Possible encoding types depend on each NVML system. The URL (i.e., "yyy" in the above description) will be set appropriately in practical uses.

2.4. Elements

The NVML is defined by using the XML (Extensible Markup Language) [2]. In this specification, we describe the NVML specific specification. Other specifications regarding to the XML, see the reference [2]. For example, although the XML reserved attribute "xml:lang" which is very useful to specify the description language, it is not explained in this specification.

In the following, on each element, the outline, basic and simple examples, the attribute, and the content are explained. In order to prescribe the values of each attribute and content, the Extended Backus-Naur Form [2] is utilized.

Note that the notation "NVML" represents the language specification while the notation "nvml" represents the root element of the NVML. In addition, the data described subject to the NVML specification is called an "NVML data", and the file containing NVML data is called an "NVML file".

2.4.1. nvml element

(1) Outline

The nvml element is the root element of the NVML and must be described. The nvml element shows the data is an NVML data.

(2) Example

```
<nvml version="0.60">
  <head> ... </head>
  <body> ... </body>
</nvml>
```

(3) Attribute

The nvml element has the following attribute:

- version
> specifies the NVML version and can take values:
> [0-9]"."[0-9][0-9]([0-9]?)
> The default value is the latest version of the NVML.

(4) Content

The nvml element has the following child elements:

- head
> The head element is defined in Section 2.4.2. The head element does not need to be described or can be described once.

- body
> The body element is defined in Section 2.4.3. The body element does not need to be described or can be described once.

These child elements must be described in the order:

> head, body.

2.4.2. head element

(1) Outline

The head element is a child element of the nvml element and does not need to be described or can be described once. The head element shows the outline of the entire navigation and guidance.

(2) Example

```
<head>
    <title>          Rainbow Town Tour </title>
    <category>       sightseeing       </category>
    <transport>      car, foot         </transport>
    <duration>       3 hours           </duration>
    <distance>       9.6 km            </distance>
    <expense>        700 yen           </expense>
    <geodetic-system> itrf             </geodetic-system>
    <note name="author" content="XYZ Tour"/>
    <info>
        <text>
            This is the Rainbow Town Tour.
        </text>
        <voice>
            Welcome to the Rainbow Town Tour!
        </voice>
```

```
                    <image src="image/rainbow-town-tour.jpg"/>
                    <note name="copyright" content="XYZ Tour"/>
                </info>
            </head>
```

(3) Attribute

The head element has no attribute.

(4) Content

The head element has the following child elements:

- **title**
    The title element is defined in Section 2.4.9. The title element does not need to be described or can be described once.
- **category**
    The category element is defined in Section 2.4.10. The category element does not need to be described or can be described any times.
- **transport**
    The transport element is defined in Section 2.4.11. The transport element does not need to be described or can be described once.
- **duration**
    The duration element is defined in Section 2.4.12. The duration element does not need to be described or can be described once.
- **distance**
    The distance element is defined in Section 2.4.13. The distance element does not need to be described or can be described once.
- **expense**
    The expense element is defined in Section 2.4.14. The expense element does not need to be described or can be described once.
- **geodetic-system**
    The geodetic-system element is defined in Section 2.4.15. The geodetic-system element does not need to be described or can be described once.
- **note**
    The note element is defined in Section 2.4.16. The note element does not need to be described or can be described any times.
- **info**
    The info element is defined in Section 2.4.8. The info element does not need to be described or can be described once.

These child elements must be described in the order:

title, category, transport, duration, distance, expense, geodetic-system, note, info.

2.4.3. body element

(1) Outline

The body element is a child element of the nvml element and does not need to be described or can be described once. The body element specifies the detail of the route navigation and the point guidances.

(2) Example

```
<body>
    <navi>   ... </navi>
    <navi>   ... </navi>
    ...
    <guide> ... </guide>
    <guide> ... </guide>
    ...
    <navi>   ... </navi>
    <guide> ... </guide>
    ...
</body>
```

(3) Attribute

The body element has no attribute.

(4) Content

The body element has the following child elements:

- navi
> The navi element is defined in Section 2.4.4. The navi element does not need to be described or can be described any times.

- guide
> The guide element is defined in Section 2.4.5. The guide element does not need to be described or can be described any times.

These child elements can be described in any order.

2.4.4. navi element

(1) Outline

The navi element is a child element of the body element and does not need to be described or can be described any times. The navi element specifies the detail of the route navigation.

(2) Example

```
<navi>
    <point>
        <name>      Tokyo Station      </name>
        <category>  Station            </category>
        <latitude>  N35.40.39.0        </latitude>
        <longitude> E139.46.18.1       </longitude>
        <address>   Chiyoda-ku, Tokyo  </address>
        <zip-code>  123-4567           </zip-code>
        <phone>     012-345-6789       </phone>
        <fax>       012-345-6789       </fax>
        <e-mail>    xxx@yyy.zzz         </e-mail>
        <url>       http://xxx.yyy.zzz </url>
    </point>
    <info duration="3min">
        <text>
            Tokyo Station
        </text>
        <voice>
```

```
                              This is Tokyo Station.
                         </voice>
                         <image src="image/tokyo-station.jpg"/>
                         <note name="copyright" content="XYZ Tour"/>
                    </info>
               </navi>


          <navi>
               <route>
                    <name>       Toumei Highway </name>
                    <category> highway            </category>
                    <means>      car              </means>
                    <duration> 10 minutes         </duration>
                    <distance> 20 km              </distance>
                    <expense>  500 yen            </expense>
               </route>
               <info duration="1min">
                    <text>
                         Toumei Highway
                    </text>
                    <voice>
                         This is Toumei Highway.
                    </voice>
                    <image src="image/tomei-highway.jpg"/>
                    <note name="copyright" content="XYZ Tour"/>
               </info>
          </navi>
```

(3) Attribute

The navi element has no attribute.

(4) Content

The navi element has the following child elements:

- **point**
  The point element is defined in Section 2.4.6. Ether one point element or one route
  element must be specified exclusively in one navi element.
- **route**
  The route element is defined in Section 2.4.7. Ether one route element or one point
  element must be specified exclusively in one navi element.
- **info**
  The info element is defined in Section 2.4.8.

These child elements must be described in the order:

point, info, *or*, route, info.

The first example in the above examples means that the information specified in the info element is output at the point specified in the point element. On the other hand, the second example means that the information described in the info element is output on the route specified in the route element.

Referring to these examples, by specifying the navi elements plural times, any navigation via point 1, point 2, and so on in a sequential order can be described as follows:

```
<navi>
    <point>
        description of the point 1
    </point>
    <info>
        description of the information
            output at the point 1
    </info>
</navi>
<navi>
    <route>
        description of the route 1-2
            from the point 1 to the point 2
    </route>
    <info>
        description of the information
            output on the route 1-2
    </info>
</navi>
<navi>
    <point>
        description of the point 2
    </point>
    <info>
        description of the information
            output at the point 2
    </info>
</navi>
<navi>
    <route>
        description of the route 2-3
            from the point 2 to the point 3
    </route>
    <info>
        description of the information
            output on the route 2-3
    </info>
</navi>
<navi>
    <point>
        description of the point 3
    </point>
    <info>
        description of the information
            output at the point 3
    </info>
</navi>
    . . .
```

Note that the route element does not need to describe the route in detail from one point to the other point (see also Section 2.4.7.). If the route element is not described or is not described in detail, the NVML system should complete the information when executing the navigation. For example, in the following case,

```
<navi>
    <point>
        description of the point 1
    </point>
</navi>
<navi>
    <route>
```

```
            description of the route A
        </route>
    </navi>
```

if the point 1 does not exist on the route A, the NVML system should complete the route from the point 1 to the route A in an appropriate manner.

In general, as child elements of the body element, any sequences of navi elements can be described. Specifically, not only the above described case, which can be described as follows by omitting the info elements for simplification,

```
<navi>
    <point> ... </point>
</navi>
<navi>
    <route> ... </route>
</navi>
<navi>
    <point> ... </point>
</navi>
<navi>
    <route> ... </route>
</navi>
    ...
```

but also the following case can also be described.

```
<navi>
    <point> ... </point>
</navi>
<navi>
    <point> ... </point>
</navi>
    ...
 <navi>
    <route> ... </route>
</navi>
<navi>
    <route> ... </route>
</navi>
    ...
```

In these incomplete cases, the NVML should complete the information on points and routes. For example, in the following case,

```
<navi>
    <route>
        description of the route A
    </route>
</navi>
<navi>
    <route>
        description of the route B
    </route>
</navi>
```

if the route A and the route B do not interchange, the NVML system should complete an

appropriate route from an appropriate point on the route A to an appropriate point on the route B.

Note that any numbers of guide elements can be inserted between navi elements and oppositely any numbers of navi elements can be inserted between guide elements. In such cases, the interpretation of the navi elements is the same as the interpretation of the cases where all the guide elements are omitted.

2.4.5. guide element

(1) Outline

The guide element is a child element of the body element and does not need to be described or can be described any times. The guide element specifies the detail of the point guidance.

(2) Example

```
<guide>
    <point area="1.0km">
        <name>          Tokyo Station          </name>
        <category>  Station               </category>
        <latitude>  N35.40.39.0           </latitude>
        <longitude> E139.46.18.1          </longitude>
        <address>   Chiyoda-ku, Tokyo  </address>
        <zip-code>  123-4567              </zip-code>
        <phone>     012-345-6789          </phone>
        <fax>       012-345-6789          </fax>
        <e-mail>    xxx@yyy.zzz           </e-mail>
        <url>       http://xxx.yyy.zzz </url>
    </point>
    <info duration="3min">
        <text>
            Tokyo Station
        </text>
        <voice>
            This is Tokyo Station.
        </voice>
        <image src="image/tokyo-station.jpg"/>
        <note name="copyright" content="XYZ Tour"/>
    </info>
</guide>
```

(3) Attribute

The guide element has no attribute.

(4) Content

The guide element specifies the detail of the point guidances. The guide element has the following child elements:

- **point**
    The point element is defined in Section 2.4.6. The point element does not need to be described or can be described once.
- **info**
    The info element is defined in Section 2.4.8.

These child elements must be described in the order:

point, info.

2.4.6. point element

(1) Outline

The point element is a child element of the navi element and the guide element. If the parent element is a navi element, ether one point element or one route element must be specified exclusively in one navi element. If the parent element is a guide element, one point element must be described in one guide element. The point element specifies the detail of the point information.

(2) Example

```
<point>
        <name>       Tokyo Station        </name>
        <category>   Station              </category>
        <latitude>   N35.40.39.0          </latitude>
        <longitude>  E139.46.18.1         </longitude>
        <address>    Chiyoda-ku, Tokyo    </address>
        <zip-code>   123-4567             </zip-code>
        <phone>      012-345-6789         </phone>
        <fax>        012-345-6789         </fax>
        <e-mail>     xxx@yyy.zzz          </e-mail>
        <url>        http://xxx.yyy.zzz   </url>
</point>
```

(3) Attribute

The point element has the following attribute:

- **area**
    is effective only when the parent element is a guide element. The area attribute specifies the area, where the information described in the info element is output, by the radius of the circle centered at the location of the point. The area attribute can take the following values:
    (0|[1-9][0-9]*)(".”[0-9]*)?("km"|"m")
    The default value is none. If no value is specified, the value should be determined by each NVML system.

(4) Content

The point element has the following child elements:

- **name**
    The name element is defined in Section 2.4.17. The name element does not need to be described or can be described once.
- **category**
    The category element is defined in Section 2.4.10. The category element does not need to be described or can be described any times.
- **latitude**
    The latitude element is defined in Section 2.4.18. The latitude element does not

need to be described or can be described once. Note that the longitude element
must also be described if the latitude element is described.

- longitude

The longitude element is defined in Section 2.4.19. The longitude element does not
need to be described or can be described once. Note that the latitude element must
also be described if the longitude element is described.

- address

The address element is defined in Section 2.4.20. The address element does not
need to be described or can be described once.

- zip-code

The zip-code element is defined in Section 2.4.21. The zip-code element does not
need to be described or can be described once.

- phone

The phone element is defined in Section 2.4.22. The phone element does not need
to be described or can be described once.

- fax

The fax element is defined in Section 2.4.23. The fax element does not need to be
described or can be described once.

- e-mail

The e-mail element is defined in Section 2.4.24. The e-mail element does not need
to be described or can be described once.

- url

The url element is defined in Section 2.4.25. The url element does not need to be
described or can be described once.

- expense

The expense element is defined in Section 2.4.14. The expense element does not
need to be described or can be described once.

- note

The note element is defined in Section 2.4.16.

These child elements must be described in the order:

name, category, latitude, longitude, address, zip-code, phone, fax, e-mail, url, expense,
note.

If the latitude element and the longitude element are specified in the point element, the NVML
system may use these information in order to determine the point location. If not specified, the
NVML system may use information of the other child elements. If the information is not enough
to determine the point location, the NVML system may deduce it as possible as the system can
do. If the system can not deduce, it should inform an error.

2.4.7. route element

(1) Outline

The route element is a child element of the navi element. Ether one point element or one route
element must be specified exclusively in one navi element. The route element specifies the
detail of the route information.

(2) Example

```
<route>
    <name>      Toumei Highway </name>
```

```
            <category> highway      </category>
            <means>    car          </means>
            <duration> 10 minutes   </duration>
            <distance> 20 km        </distance>
            <expense>  500 yen      </expense>
        </route>
```

(3) Attribute

The route element has no attribute.

(4) Content

The route element has the following child elements:

- name
> The name element is defined in Section 2.4.17. The name element does not need to be described or can be described once.

- category
> The category element is defined in Section 2.4.10. The category element does not need to be described or can be described any times.

- number
> The number element is defined in Section 2.4.26. The number element does not need to be described or can be described once.

- means
> The means element is defined in Section 2.4.27. The means element does not need to be described or can be described once.

- duration
> The duration element is defined in Section 2.4.12. The duration element does not need to be described or can be described once.

- distance
> The distance element is defined in Section 2.4.13. The distance element does not need to be described or can be described once.

- expense
> The expense element is defined in Section 2.4.14. The expense element does not need to be described or can be described once.

- note
> The note element is defined in Section 2.4.16. The note element does not need to be described or can be described any times.

- info
> The info element is defined in Section 2.4.8. The info element does not need to be described or can be described once.

These child elements must be described in the order:

name, category, number, means, duration, distance, expense, note, info.

2.4.8. info element

(1) Outline

The info element is a child element of the head element, the navi element, and the guide element and does not need to be described or can be described once. The info element specifies the detail of the output information.

If the parent element is a head element, the timing of outputting the information described in the info element should be determined by each NVML system. If the parent element is a navi element, the information described in the info element is output when the current location arrives at the point specified in the point element that is the brother element of the info element. If the parent element is a guide element, the information described in the info element is output when the current location comes in the area around the point specified in the point element which is the brother element of the info element. See the following attribute subsection about the detail.

Notice that in some cases, navi elements and guide elements are possible to desire to output some information at the same time. In such cases, it depends on each NVML system whether the two (or more) information will be output at the same time or one comes before the other.

(2) Example

```
<info duration="3min">
    <text>
        Tokyo Station
    </text>
    <voice>
        This is Tokyo Station.
    </voice>
    <image src="image/tokyo-station.jpg"/>
    <note name="copyright" content="XYZ Tour"/>
</info>
```

(3) Attribute

The info element has the following attributes:

- **delay**
    specifies the beginning time that the output of the information begins, and can take the values as follows:
        ("+"|"-")?
        ((0|[1-9][0-9]*)("."[0-9]*)?"h")?
        ((0|[1-9][0-9]*)("."[0-9]*)?"min")?
        ((0|[1-9][0-9]*)("."[0-9]*)?"s")?
    The cases of no sign and with "+" sign mean a delay from the default beginning time, and the case with "-" sign means a gain from the default beginning time.

    The default beginning time is set as follows:

        - **In the case that the parent element is a head element,**
            it should be determined by each NVML system.
        - **In the case that the parent element is a navi element,**
            it is the time arriving at the point specified in the brother point element. In this case, if the beginning time becomes later than the default beginning time of the next navi element, or the beginning time becomes earlier than the default beginning time of the previous navi element, the output of the information becomes invalid.
        - **In the case that the parent element is a guide element,**
            it is the time coming in the area specified in the brother point element. Note that in this case the default beginning time is reset every time that the current location comes in the area.

The default value is none. If no value is specified, the value should be determined by each NVML system.

```
case: delay >= 0
  -----------|----------------
              default beginning time
              |---->|
               delay
                    |--------------->|
                            duration

case: delay <= 0
  -----------|----------------
              default beginning time
         |<----|
          delay
         |--------------->|
              duration
```

- duration
specifies the duration time to output the information and can take values:

```
(((0|[1-9][0-9]*)("."[0-9]*)?"h")?
((0|[1-9][0-9]*)("."[0-9]*)?"min")?
((0|[1-9][0-9]*)("."[0-9]*)?"s")?)|
"full-time"
```

The value "full-time" means follows:
- **In the case that the parent element is a head element,**
    the NVML system determine how the information is output.
- **In the case that the parent element is a navi element,**
    the output of the information is kept until the next navi element becomes valid.
- **In the case that the parent element is a guide element,**
    the output of the information is kept until the current location gets out of the area specified by the brother point element.

The default value is none. If no value is specified, the value should be determined by each NVML system.

- times
is effective only when the parent element is a guide element. If there exists a point element as a brother element, the information specified in the info element is output certain times, which is specified by the times attribute, when the current location comes in the area specified by the area attribute. The times attribute takes the values:

```
(0|[1-9][0-9]*)|"any-times"
```

The value "any-times" means that the information is output any times. The default value is none. If no value is specified, the value should be determined by each NVML system.

(4) Content

The info element has the following child elements:

- text
The text element is defined in Section 2.4.28. The text element does not need to be described or can be described once.

- voice

The voice element is defined in Section 2.4.29. The voice element does not need to be described or can be described once.

- **image**

The image element is defined in Section 2.4.30. The image element does not need to be described or can be described once.

- **note**

The note element is defined in Section 2.4.16.

These child elements must be described in the order:

text, voice, image, note.

2.4.9. title element

(1) Outline

The title element is a child element of the head element and does not need to be described or can be described once. The title element specifies the title of the navigation.

(2) Example

```
<title> Rainbow Town Tour </title>
```

(3) Attribute

The title element has no attribute.

(4) Content

The title element describes the title of the entire navigation and guidances. The title element can take any text character sequences. Each character is any character permitted to use in the XML specification [2].

2.4.10. category element

(1) Outline

The category element is a child element of the head element, the point element, and the route element, and does not need to be described or can be described any times. The category element describes the category of the information that the parent element represents.

(2) Example

```
<head>
    <category> sightseeing </category>
</head>

<point>
    <category> restaurant </category>
    <category> Italian    </category>
</point>

<route>
```

```
            <category> highway </category>
        </route>
```

(3) Attribute

The category element has no attribute.

(4) Content

The category element describes the category of the information that the parent element represents. In the case that the parent element is a head element, the category element describes the category of the entire navigation and guidances of the NVML. In the case that the parent element is a point element, the category element describes the category of the point. In the case that the parent element is a route element, the category element describes the category of the route. The method of the categorization and the detail of the categories are not specified in this specification.

The category element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The category element has no default value.

2.4.11. transport element

(1) Outline

The transport element is a child element of the head element and does not need to be described or can be described once. The transport element describes the means of transportation which are used during the navigation.

(2) Example

```
            <transport> car, foot </transport>
```

(3) Attribute

The transport element has no attribute.

(4) Content

The transport element describes the means of transportation which are used during the navigation. The method of the categorization and the detail of the categories are not specified in this specification.

The transport element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The transport element has no default value.

2.4.12. duration element

(1) Outline

The duration element is a child element of the head element and the route element and does not need to be described or can be described once. The duration element describes the time duration required to navigate the course or the route.

(2) Example

```
<duration> 3 hours </duration>
```

(3) Attribute

The duration element has no attribute.

(4) Content

The duration element describes the time duration required to navigate the course or the route. The method of the categorization and the detail of the categories are not specified in this specification.

The duration element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The duration element has no default value.

The use of the information of the duration element depends on each NVML system. For example, if a system searches and finds a route considering the traffic congestion and then estimates the time duration, the system may use the estimated time duration rather than that of the duration element.

2.4.13. distance element

(1) Outline

The distance element is a child element of the head element and the route element and does not need to be described or can be described once. The distance element specifies the distance of the course or the route.

(2) Example

```
<distance> 9.6km </distance>
```

(3) Attribute

The distance element has no attribute.

(4) Content

The distance element describes the distance of the course or the route. The method of the categorization and the detail of the categories are not specified in this specification.

The distance element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The distance element has no default value.

The use of the information of the distance element depends on each NVML system. For example, if a system searches and finds a route considering the traffic congestion and then estimates the distance, the system may use the estimated distance rather than that of the distance element.

2.4.14. expense element

(1) Outline

The expense element is a child element of the head element, the point element, and the route element and does not need to be described or can be described once. The expense element describes the required expense, such as admission fee, toll, and bus/train fare.

(2) Example

```
<expense> 700 yen </expense>
```

(3) Attribute

The expense element has no attribute.

(4) Content

The expense element describes the required expense, such as admission fee, toll, and bus/train fare. The method of the categorization and the detail of the categories are not specified in this specification.

The expense element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The expense element has no default value.

The use of the information of the expense element depends on each NVML system. For example, if a system searches and finds a route considering the traffic congestion and then estimates the expense, the system may use the estimated expense rather than that of the expense element.

2.4.15. geodetic-system element

(1) Outline

The geodetic-system element is a child element of the head element and does not need to be described or can be described once. The geodetic-system element specifies the geodetic system. The values of the latitude element and the longitude element of each point element are represented in the reference frame of this geodetic system.

(2) Example

```
<geodetic-system> itrf </geodetic-system>
```

(3) Attribute

The geodetic-system element has no attribute.

(4) Content

The geodetic-system element specifies the geodetic-system in which the latitude and the longitude of each point are represented. The geodetic-system element can take the values as

follows:

```
("itrf"|"wgs84")
```

and any text character sequences where each character is any character permitted to use in the XML specification [2]. The value "itrf" means the International Terrestrial Reference Frame (ITRF) [3] and the value "wgs84" means the World Geodetic System 1984 (WGS 84) [4]. The default value is "itrf".

2.4.16. note element

(1) Outline

The note element is a child element of the head element, point element, route element, and info element and does not need to be described or can be described any times. Users can describe any notes in note elements. The note element is similar to the macro element of the Synchronized Multimedia Integration Language (SMIL) [5].

(2) Example

```
<note name="author" content="XYZ Tour"/>
```

(3) Attribute

The note element has the following attributes.

- **name**
    describes the name of the note. The name attribute can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The name attribute has no default value.
- **content**
    describes the content of the note. The content attribute can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The content attribute has no default value.

(4) Content

The note element has no content.

2.4.17. name element

(1) Outline

The name element is a child element of the point element and the route element and does not need to be described or can be described once. The name element describes the name of the point or the route.

(2) Example

```
<point>
    <name> Tokyo Station </name>
```

```
            </point>

            <route>
                <name> Toumei Highway </name>
            </route>
```

(3) Attribute

The name element has no attribute.

(4) Content

The name element describes the name of the point or the route. The name element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The name element has no default value.

2.4.18. latitude element

(1) Outline

The latitude element is a child element of the point element and does not need to be described or can be described once. If the latitude element is described, then the longitude element must also be described. The latitude element specifies the latitude of the point.

(2) Example

```
            <latitude> N35.37.28.72 </latitude>
```

(3) Attribute

The latitude element has no attribute.

(4) Content

The latitude element specifies the latitude of the point. The latitude element can take the following latitude values in the reference frame of the geodetic system specified in the geodetic-system element of the head element.

```
("N"|"S")
  ((0?[0-9])|([1-8][0-9])|90)"."
  ((0?[0-9])|([1-5][0-9]))"."
  ((0?[0-9])|([1-5][0-9]))
  ("."[0-9]*)?
```

The terms "N" and "S" represent north and south latitudes respectively. The latitude element has no default value.

If the latitude element and the longitude element are specified in the point element, the NVML system may use these information to determine the point location. If not specified, the system may use information of the other child elements. If the information is not enough to determine the point location, the system may deduce it as possible as the system can do. If the system can not deduce, it should inform an error.

2.4.19. longitude element

(1) Outline

The longitude element is a child element of the point element and the longitude element does not need to be described or can be described once. If the longitude element is described, then the latitude element must also be described. The longitude element specifies the longitude of the point.

(2) Example

```
<longitude> E139.37.52.128 </longitude>
```

(3) Attribute

The longitude element has no attribute.

(4) Content

The longitude element specifies the longitude of the point. The longitude element can take the following longitude values in the reference frame of the geodetic system specified in the geodetic-system element of the head element.

```
("E"|"W")
  ((0?0?[0-9])|(0?[1-9][0-9])|(1[1-7][0-9])|180)"."
  ((0?[0-9])|([1-5][0-9]))"."
  ((0?[0-9])|([1-5][0-9]))
  ("."[0-9]*)?
```

The terms "E" and "W" represent east and west longitudes respectively. The longitude element has no default value.

If the longitude element and the latitude element are specified in the point element, the NVML system may use these information in order to determine the point location. If not specified, the system may use information of the other child elements. If the information is not enough to determine the point location, the system may deduce it as possible as the system can do. If the system can not deduce, it should inform an error.

2.4.20. address element

(1) Outline

The address element is a child element of the point element and the address element does not need to be described or can be described once. The address element specifies the address of the point.

(2) Example

```
<address> Chiyoda-ku, Tokyo </address>
```

(3) Attribute

The address element has no attribute.

(4) Content

The address element specifies the address of the point. The address element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The address element has no default value.

2.4.21. zip-code element

(1) Outline

The zip-code element is a child element of the point element and does not need to be described or can be described once. The zip-code element specifies the zip-code of the point.

(2) Example

```
<zip-code> 123-4567 </zip-code>
```

(3) Attribute

The zip-code element has no attribute.

(4) Content

The zip-code element specifies the zip-code of the point. The zip-code element can take the following values:

```
([0-9]|"-")*
```

The zip-code element has no default value.

2.4.22. phone element

(1) Outline

The phone element is a child element of the point element and does not need to be described or can be described once. The phone element specifies the phone number related to the point.

(2) Example

```
<phone> 012-345-6789 </phone>
```

(3) Attribute

The phone element has no attribute.

(4) Content

The phone element specifies the phone no. related to the point. The phone element can take the following values:

```
( [0-9] | "+" | "-" ) *
```

The phone element has no default value.

2.4.23. fax element

(1) Outline

The fax element is a child element of the point element and does not need to be described or can be described once. The fax element specifies the fax number related to the point.

(2) Example

```
<fax> 012-345-6789 </fax>
```

(3) Attribute

The fax element has no attribute.

(4) Content

The fax element specifies the fax no. related to the point. The fax element can take the following values:

```
( [0-9] | "+" | "-" ) *
```

The fax element has no default value.

2.4.24. e-mail element

(1) Outline

The e-mail element is a child element of the point element and does not need to be described or can be described once. The e-mail element specifies the e-mail related to the point.

(2) Example

```
<e-mail> xxx@yyy.zzz </e-mail>
```

(3) Attribute

The e-mail element has no attribute.

(4) Content

The e-mail element specifies the e-mail address related to the point. The e-mail element can take the following values:

```
( [0-9] | [a-z] | [A-Z] | "@" | "." ) *
```

More exactly speaking, the value is subject to the RFC (Request for Comments) 822 [6]. The e-mail element has no default value.

2.4.25. url element

(1) Outline

The url element is a child element of the point element and does not need to be described or can be described once. The url element specifies the url of the feature located at the point.

(2) Example

```
<url> http://xxx.yyy.zzz </url>
```

(3) Attribute

The url element has no attribute.

(4) Content

The url element specifies the URL (Uniform Resource Locator) related to the point. The url element can take the following values:

```
( [0-9] | [a-z] | [A-Z] | ":" | "/" | "." )*
```

More exactly speaking, the value is subject to the RFC (Request for Comments) 1738 [7]. The url element has no default value.

2.4.26. number element

(1) Outline

The number element is a child element of the route element and does not need to be described or can be described once. The number element specifies the number of the route.

(2) Example

```
<number> 123 </number>
```

(3) Attribute

The number element has no attribute.

(4) Content

The number element specifies the number of the route. The number element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The number element has no default value.

2.4.27. means element

(1) Outline

The means element is a child element of the route element and does not need to be described or can be described once. The means element specifies the means of the moving.

(2) Example

```
<means> car </means>
```

(3) Attribute

The means element has no attribute.

(4) Content

The means element specifies the means of the moving. The means element can take the following values:

```
("foot"|
 "bicycle"|
 "motorcycle"|
 "car"|
 "bus"|
 "train"|
 "plane"|
 "ship"|
 "others")
```

The means element has no default value. If the system is a car navigation system, the default value may be "car", and if the system is a cellar phone, the default value may be "foot".

2.4.28. text element

(1) Outline

The text element is a child element of the info element and does not need to be described or can be described once. The text element describes the information by texts.

(2) Example

```
<text>
    Tokyo Station
</text>
```

(3) Attribute

The text element has no attribute.

(4) Content

The text element describes the information by texts. The text element can take any text character sequences. Each character is any character permitted to use in the XML specification

[2]. The text element has no default value.

2.4.29. voice element

(1) Outline

The voice element is a child element of the info element and does not need to be described or can be described once. The voice element describes the information by voice.

(2) Example

```
<voice src="voice/tokyo-station.avi"/>

<voice>
    This is Tokyo Station.
</voice>
```

(3) Attribute

The voice element has the following attribute:

- **src**

  specifies the name of the voice file to be output as information. Acceptable voice file types depend on each NVML system.

(4) Content

The voice element describes the information by voice. The voice element can take any text character sequences. Each character is any character permitted to use in the XML specification [2]. The voice element has no default value. If both the src attribute and the content are specified, the NVML system should use the src attribute first.

2.4.30. image element

(1) Outline

The image element is a child element of the info element and does not need to be described or can be described once. The image element specifies the information by image.

(2) Example

```
<image src="image/tokyo-station.jpg"/>
```

(3) Attribute

The image element has the following attribute:

- **src**

  specifies the name of the image file to be output as information. Acceptable image file types depend on each NVML system.

(4) Content

The image element has no content.

3. Concluding Remarks

The "NVML" (NaVigation Markup Language) is a markup language for describing the navigation information such as locations of points and route information. The NVML enables us to use a navigation service at any time and anywhere, and it will also be used in other services such as a guide service for sightseeing, a travel planning service, and a publishing service for tourists.

The current NVML can not describe in detail or perfectly various information of points, routes, or features. These information will be categorized and described by each special society. The nature of the NVML is to describe the sequences consisting of time, point, and information, in other words, navigation and guidance information. By inputting NVML data to systems, information will be output depending on the time and the point.

As future developments of NVML, we should consider the following issues:

- **Time Concept**
    The time concept, for example, starting time, arriving time, or staying time, should be included.
- **Relative Time and Point Representations**
    Relative time and point representations should be permitted for convenience. For example, (output some information)
        - 10 minutes before arriving at the next point;
        - 10 minutes after starting from the current point;
        - 10 km before arriving at the next point; or
        - 10 km after starting from the current point.
- **Values and Units**
    Values and units of attributes and contents, should be prescribed more precisely. For example, duration, distance, expense, and so on. Note that this problem is related to the multilingual problem.
- **Referring Functionality**
    For example, information on points and routes should be refereed from other or outer resources such as external files or webs. This referring functionality will be realized by using the entity functionality of the XML [2] or more generally by using the XML Linking Language (XLink) [8] and the XML Pointer Language (XPointer) [9].
- **Resource Description Framework (RDF)**
    The current version of NVML is defined by using the Document Type Definition (DTD). A future another way is to use the Resource Description Framework (RDF) [11] since the NVML contains many meta data.
- **Integration with Other Markup Languages**
    The info element may be integrated with or substituted by some specialized markup language, for example, the Synchronized Multimedia Integration Language (SMIL) [5]. Such integration or substitution will be realized by using the namespaces functionality [10].
- **Fundamentality and Flexibility**
    The NVML should be compact and fundamental to represent times, points, and information. However, the NVML should also have flexibility so as to be used in many applications.

4. References

[1] N. Borenstein and N. Freed, "MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies," RFC (Request for Comments): 1521, Sep. 1993. ftp://ftp.isi.edu/in-notes/rfc1521.txt

[2] "Extensible Markup Language (XML) 1.0," World Wide Web Consortium (W3C) Recommendation, REC-xml-19980210, Feb. 2, 1998. http://www.w3.org/TR/1998/REC-xml-19980202

[3] "International Terrestrial Reference Frame (ITRF)," International Earth Rotation Service (IERS), http://hpiers.obspm.fr/

[4] "WGS 84 - World Geodetic System 1984," European Organization for the Safety of Air Navigation (EUROCONTROL), http://www.wgs84.com/

[5] "Synchronized Multimedia Integration Language (SMIL) 1.0 Specification," World Wide Web Consortium (W3C) Recommendation, REC-smil-19980615, Jun. 15, 1998. http://www.w3.org/TR/1998/REC-smil-19980615

[6] David H. Crocker, "Standard for the Format of ARPA Internet Text Messages," RFC (Request for Comments): 822, Aug. 13, 1982. http://www.morito.mgmt.waseda.ac.jp/~tom/RFC/rfc822.txt

[7] T. Berners-Lee, L. Masinter, and M. McCahill, "Uniform Resource Locators (URL)," RFC (Request for Comments): 1738, Dec. 1994. http://www.morito.mgmt.waseda.ac.jp/~tom/RFC/rfc1738.txt

[8] "XML Linking Language (XLink)," World Wide Web Consortium (W3C) Working Draft, WD-xlink-19980303, Mar. 3, 1998. http://www.w3.org/TR/1998/WD-xlink-19980303

[9] "XML Pointer Language (XPointer)," World Wide Web Consortium (W3C) Working Draft, WD-xptr-19980303, Mar. 3, 1998. http://www.w3.org/TR/1998/WD-xptr-19980303

[10] "Namespaces in XML," World Wide Web Consortium (W3C) Recommendation, REC-xml-names-19990114, Jan. 14, 1999. http://www.w3.org/TR/1999/REC-xml-names-19990114

[11] "Resource Description Framework (RDF) Model and Syntax Specification," World Wide Web Consortium (W3C) Recommendation, REC-rdf-syntax-19990222, Feb. 22, 1999. http://www.w3.org/TR/1999/REC-rdf-syntax-19990222

5. Appendices

5.1. DTD

The XML Document Type Definition (DTD) of the NaVigation Markup Language (NVML) is shown below:

```
<!-- nvml.dtd -->
<!-- Document Type Definition (DTD) of NVML (Draft) -->
<!-- June 30, 1999 -->


<!ELEMENT        nvml                    (head?,body?)>
   <!ATTLIST     nvml                    version        CDATA        #IMPLIED>
<!ELEMENT        head                    (title?,category*,transport?,
                                         duration?,distance?,expense?,
                                         geodetic-system?,note*,info?)>
<!ELEMENT        body                    (navi|guide)*>
<!ELEMENT        navi                    ((point|route),info?)>
<!ELEMENT        guide                   (point,info?)>
<!ELEMENT        point                   (name?,category*,(latitude,longitude)?,
                                         address?,zip-code?,
                                         phone?,fax?,e-mail?,url?,
```

```
                                                     expense?,
                                                     note*)>
              <!ATTLIST          point          area         CDATA          #IMPLIED>
          <!ELEMENT          route          (name?,category*,number?,means?,
                                             duration?,distance?,expense?,
                                             note*)>
          <!ELEMENT          info           (text?,voice?,image?,note*)>
            <!ATTLIST        info           delay      CDATA      #IMPLIED
                                            duration   CDATA      #IMPLIED
                                            times      CDATA      #IMPLIED>

          <!ELEMENT   title          (#PCDATA)>
          <!ELEMENT   category       (#PCDATA)>
          <!ELEMENT   transport      (#PCDATA)>
          <!ELEMENT   duration       (#PCDATA)>
          <!ELEMENT   distance       (#PCDATA)>
          <!ELEMENT   expense        (#PCDATA)>
          <!ELEMENT   geodetic-system  (#PCDATA)>
          <!ELEMENT   note           EMPTY>
            <!ATTLIST  note          name       CDATA      #IMPLIED
                                     content    CDATA      #IMPLIED>

          <!ELEMENT   name           (#PCDATA)>
          <!ELEMENT   latitude       (#PCDATA)>
          <!ELEMENT   longitude      (#PCDATA)>
          <!ELEMENT   address        (#PCDATA)>
          <!ELEMENT   zip-code       (#PCDATA)>
          <!ELEMENT   phone          (#PCDATA)>
          <!ELEMENT   fax            (#PCDATA)>
          <!ELEMENT   e-mail         (#PCDATA)>
          <!ELEMENT   url            (#PCDATA)>

          <!ELEMENT   number         (#PCDATA)>
          <!ELEMENT   means          (#PCDATA)>

          <!ELEMENT   text           (#PCDATA)>
          <!ELEMENT   voice          (#PCDATA)>
            <!ATTLIST  voice         src        CDATA      #IMPLI
          <!ELEMENT   image          EMPTY>
            <!ATTLIST  image         src        CDATA      #IMPLI
```

5.2. Quick Table

The quick table of the NaVigation Markup Language (NVML) is shown below:

```
--------------------------------------------------------------------------------
Element            Attribute        Times      Examples
                                    (#1)
--------------------------------------------------------------------------------
nvml
|                  version          0/1        "0.60"
+-head                              0/1        -
| +-title                          0/1        XXX-Tour
| +-category                       0-         sightseeing
| +-transport                      0/1        car, foot
| +-duration                       0/1        4 hours 10 minutes
| +-distance                       0/1        32.1 km
| +-expense                        0/1        3,000 yen
| +-geodetic-system                0/1        itrf
| |                                           wgs84
| +-note                           0-         -
| |                name            0/1        "last modified"
```

```
|  |             content         0/1      "Jun. 30, 1999"
|  +-info                        0/1         -
|  |             delay           0/1      "1h23min45s"
|  |             duration        0/1      "1h23min45s"
|  +-text                        0/1      This is XXX-Tour.
|  +-voice                       0/1      Welcome to XXX-Tour!
|  |             src             0/1      "welcome.avi"
|  +-image                       0/1         -
|  |             src             0/1      "welcome.jpg"
|  +-note                        0-          -
|                name            0/1      "copyright"
|                content         0/1      "XXX Tour"
+-body                           0/1         -
   +-navi                        0-          -
   |  +-point                    0/1         -
   |  |  +-name                  0/1      Tokyo Station
   |  |  +-category              0-       station
   |  |  +-latitude              0/1      N35.40.39.0
   |  |  +-longitude             0/1      E139.46.18.1
   |  |  +-address               0/1      Chiyoda-ku, Tokyo
   |  |  +-zip-code              0/1      012-3456
   |  |  +-phone                 0/1      012-345-6789
   |  |  +-fax                   0/1      012-345-6789
   |  |  +-e-mail                0/1      xxx@yyy.zzz
   |  |  +-url                   0/1      http://xxx.yyy.zzz
   |  |  +-expense               0/1      1,000 yen
   |  |  +-note                  0-          -
   |  |             name         0/1      "access"
   |  |             content      0/1      "JR, subway, bus, car"
   |  +-route                    0/1         -
   |  |  +-name                  0/1      Toumei Highway
   |  |  +-category              0-       highway
   |  |  +-number                0/1      123
   |  |  +-means                 0/1      foot
   |  |  |                                bicycle
   |  |  |                                motorcycle
   |  |  |                                car
   |  |  |                                bus
   |  |  |                                train
   |  |  |                                plane
   |  |  |                                ship
   |  |  |                                others
   |  |  +-duration              0/1      1 hour 40 minutes
   |  |  +-distance              0/1      12.3 km
   |  |  +-expense               0/1      1,000 yen
   |  |  +-note                  0-          -
   |  |             name         0/1      "max speed"
   |  |             content      0/1      "100 km/h"
   |  +-info                     0/1         -
   |  |             delay        0/1      "1h23min45s"
   |  |             duration     0/1      "1h23min45s"
   |  +-text                     0/1      Tokyo Station
   |  +-voice                    0/1      This is Tokyo Station.
   |  |             src          0/1      "tokyo-station.avi"
   |  +-image                    0/1         -
   |  |             src          0/1      "tokyo-station.jpg"
   |  +-note                     0-          -
   |                name         0/1      "copyright"
   |                content      0/1      "XYZ Studio"
   +-guide                       0-          -
      +-point                    1           -
      |  |          area         0/1      "1.0km"
      |  +-name                  0/1      Tokyo Station
      |  +-category              0-       station
```

```
        |   +-latitude              0/1      N35.40.39.0
        |   +-longitude             0/1      E139.46.18.1
        |   +-address              '0/1      Chiyoda-ku, Tokyo
        |   +-zip-code              0/1      012-3456
        |   +-phone                 0/1     ·012-345-6789
        |   +-fax                   0/1      012-345-6789
        |   +-e-mail                0/1      xxx@yyy.zzz
        |   +-url                   0/1      http://xxx.yyy.zzz
        |   +-expense               0/1      1,000 yen
        |   +-note                  0-         -
        |           name            0/1      "access"
        |           content         0/1      "JR, subway, bus, car"
        +-info                      0/1        -
        |       delay               0/1      "1h23min45s"
        |       duration            0/1      "1h23min45s"
        |       times               0/1      "1"
        |                                    "any-times"
        +-text                      0/1      Tokyo Station
        +-voice                     0/1      This is Tokyo Station.
        |       src                 0/1      "tokyo-station.avi"
        +-image                     0/1        -
        |       src                 0/1      "tokyo-station.jpg"
        +-note                      0-         -
                name                0/1      "copyright"
                content             0/1      "XYZ Studio"
------------------------------------------------------------------------
(#1) 0- : 0 or more
     0/1: 0 or 1
------------------------------------------------------------------------
```

THIS PAGE BLANK (USPTO)